University of Glasgow | School of Computing Science

Honours Individual Project Dissertation

Evaluating Fair Learning-to-Rank Strategies for Wikipedia Articles

Stamatis Theocharous

September 14, 2022

Abstract

This dissertation investigates the bias in the Wikipedia search engine, where editors receive articles with which are primarily based on certain dominant characteristics, that can result in under-exposed protected groups receiving an unfair exposure. The goal of this study is to develop and implement seven strategies that will use learning to rank to decrease this unfairness without reducing the relevance of articles assigned to editors from a vertical ranked list. The study employs feature selection and feature boosting techniques during the post-processing phase of machine learning to promote fairness. Although the results did not yield statistical significance, the study identified a single best algorithm for boosting fairness and detecting which features have a higher importance in the learning to rank model using the Trec-Fair 2022 dataset.

Acknowledgements

I would like to express my heartfelt gratitude to Dr. Graham McDonald and my family for their unwavering support and guidance throughout the duration of this dissertation. Their valuable assistance and constructive feedback have been essential in shaping the development of this thesis.

Education Use Consent

I hereby grant my permission for this project to be stored, distributed and shown to other University of Glasgow students and staff for educational purposes. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Signature: Stamatis Theocharous Date: 30 March 2023

Contents

1 Introduction

The first search engine was created in the 1990s, named Archie which aimed to find files hosted on a FTP (file transfer protocol) server and it was the leading step in the development of modern era search engines as stated by Livemint (2018). Search Engines now like google and Wikipedia have to go through millions or billions of documents in order to retrieve information based on a query. This problem led to the formation of ranking systems that use machine learning to learn which documents are relevant to a user based on several information and rank them according to their relevance. These are called Learning to Rank (LTR) systems.

1.1 Motivation

These ltr systems are considered very effective in creating search engines due to their ability to optimally retrieve the most relevant information for the user from a huge dataset, thus marking their importance in modern search engines. However, they suffer from a significant flaw that has become the focus of recent research in the ltr field. Such systems only consider relevance when ranking documents and fail to account for fairness, as demonstrated by Singh and Joachims (2018). We identify fairness as the equal exposure of all groups within the top-k documents in the ranking. The reason that we use the top-k documents is due to the position bias where it means that in a vertical ranked list of documents users are much less likely that will look lower at the list. For example, a hiring system that searches for candidates for the role of software engineers and due to the much higher amount of males in this job it is much more likely that the top-k candidates will be males than females in a vertical list. Therefore, there is a biased candidate selection as female software engineers are under exposed to the hirring managers meaning that female software engineers are less likely to find a job in that career thus causing a real world issue. This issue has been highlighted by the Text Retrieval Conference (Trec) which is an organisation majorly sponsored by the United States Government, through the National Institute of Standards and Technology (NIST). During the last 4 years they have provided to researchers a platform with necessary and standardised tools such as datasets, metrics and evaluation techniques to develop fair strategies, to promote the fairer exposure of protected groups and try to minimise these real-life issues. This program has been named as Trec-Fair.

In particular, Wikipedia has been identified by Redi et al. (2020) as an unfair system due to the highly uneven distribution of groups in its dataset. For instance, as we can see from **figure 1.2** which is a color map of active Wikipedia editors, the majority of articles on Wikipedia are created/edited by editors in North America and Europe, which means that Wikipedia editors are more likely to come across articles from these areas when using the Wikipedia search engine to find articles to edit. This can result in a higher number of articles to be edited by people in North America and Europe, and a lower number from other continents such as Africa. This example follows the Natural Bias problem where editors will only be recommended articles largely by the most famous regions, due to there much higher exposure. This will lead to a scenario of documents with higher exposure whereby such an exposure will continue to be ranked higher and higher in the ranking, creating a systematic bias, thus leading to a number of real-life issues. Such issues which include, for example, countries that have higher exposure in Wikipedia articles and have resulted in higher education with more users gaining knowledge from the free

encyclopedia resulting in higher economic status, as stated by Pedreshi et al. (2008). Meaning, that the unfair nature of the Wikipedia search engine has real life consequences. Therefore, in this dissertation we aim to propose several strategies that will allow for these under-exposed articles to have a higher exposure and improve the overall fairness of the Wikipedia search engine. Our strategies will be implemented during the post-processing phase of learning where they will act as re-ranking strategies based on some scores that aim to improve fairness. We reference these scores as fairness as the ltr performs re-re-ranking and learns complex relationships with the use feature scores. As shown in **Figure 1.1** we are proposing in integrating our strategies during the post-processing phase where our fairness features, will be combined with the relevance features after the top 1000 documents have been retrieved by a BM25 retrieval model and then applied to the ltr and act re rankers to retrieve the most relevant documents.

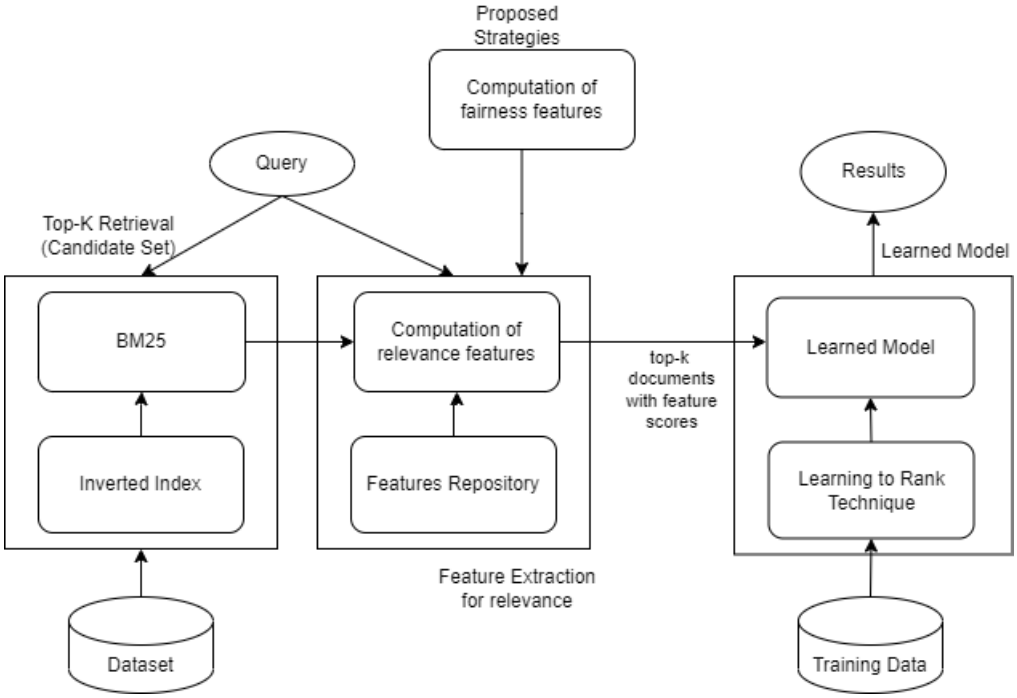


***Figure** 1.1: The figure shows the process of creating a ltr model and where we are proposing in integrating our strategies in the model. The process starts by filtering our the irrelvant documents first and resulting with the top-k documents according to a query. Then proceed with the feature creation of based on other information retrieval models and combine these scores for each document in the top-k list. Then we compute our fairness scores and integrate them into the list of scores computed by the information retrieval models. These scores are used by the trained ltr algorithm as re-rankers to create a ranked list with fairer exposure of protected groups. This figure was inspired by Tonellotto et al. (2013)*

1.2 Aims

This dissertation aims to address the issue of unfairness in the Wikipedia search engine by developing a number of ltr approaches that will increase the exposure protected groups and provide Wikipedia editors a greater fairer distribution of articles in a vertical ranking of documents having the most relevant document at the top and the least relevance at the bottom. These aims are synced with goals of Trec-Fair 2022 which will be discussed in more detail in **Chapter** 2 which are broken down into these tasks:

- Develop a set of strategies and integrate them into LTR models to incorporate fairness

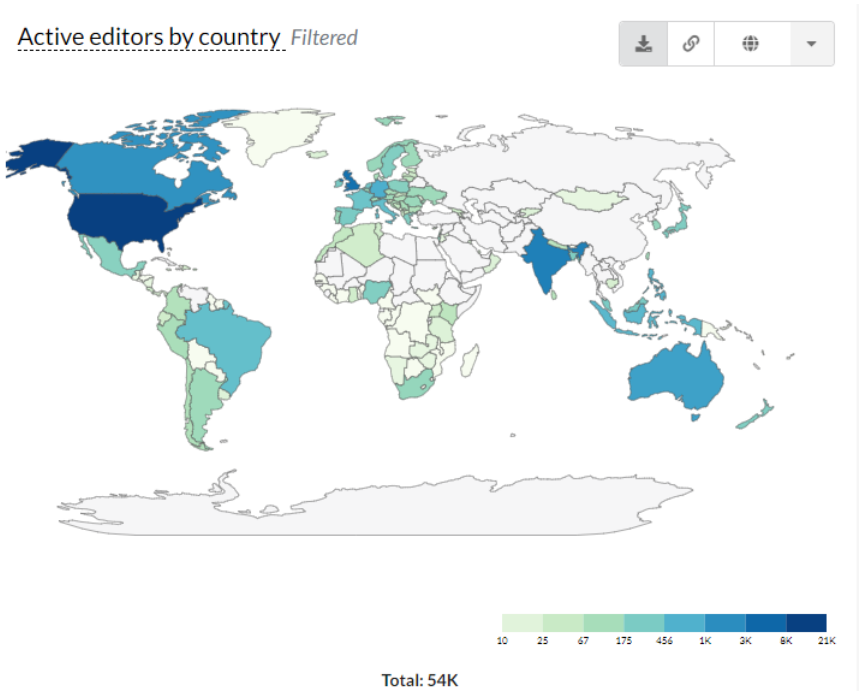


Figure 1.2: *This color map shows the countries with most active Wikipedia editors per month. The darker the shade of blue, the higher the number of editors. Therefore, this graph indicates that the USA and the UK have the most active editors and the 2 continents with the most active editors being North America and Europe. Source: Wikimedia Foundation (2021)*

- Using statistical analysis on a sequence of rankings with different fairness features to check if there is a significant effect in the average fairness and relevance scores in all of the sequence rankings

1.3 Dissertation Structure

The **introductory** chapter outlines the importance of fairness in LTR systems and defines the objectives of this dissertation. The remainder of the dissertation will provides the necessary background needed to understand the fairness algorithms designed and implemented along with their evaluations. The dissertation structure is as follows:

- **Chapter 2** outlines how previous work has been ingested into the LTR implementations and examines what fairness is
- **Chapter 3** describes the Proposed strategies and their implementation.
- **Chapter 4** discusses the necessary tools, equipment and design choices needed to implement the experiment.
- **Chapter 6** provides an evaluation of the new algorithms, including relevance, fairness scores and significance testing.
- **Chapter 7** summarises this dissertation, examines any future work which could be carried out and provides an overall reflection of the project.

2 Background

This chapter begins by exploring the definitions of fairness and how can fairness be implemented in ltr with examples of existing implementations and techniques. Then Trec-Fair is discussed, tracing its evolution from Trec-Fair 2020 to Trec-Fair 2022 and the Trec-Fair datasets. Lastly, there is a brief discussion on metrics used in ltr with an overview of how everything is connected.

2.1 Fairness

2.2 Types of Fairness

Fairness in machine learning has been an important focus of research in the recent years as new state of the art approaches are constantly being created to address the issues raised in the **Chapter 1**. Hence, to address the issue of fairness we first have to define it. We have discovered through literature that there is currently no apparent universal definition for fairness in Information Retrieval (IR) or ML. However most of the definitions can be clustered into whether their objective is to treat different groups of individuals, with common characteristics such as demographics similarly (group fairness), or treat similar individuals similarly (individual fairness).

The individual fairness goal is to provide personalised rankings for each user, while ensuring that the results are fair with respect to sensitive features. By sensitive features we mean the under-exposed groups that are used as information in the ml. For example, in the Wikipedia editor scenario, such ltr models would use the editors previously edited articles as data to predict what articles the editor would like to edit. This approach might discriminate against some sensitive features such as gender or occupation. On the other hand in the case that all of an editors previous articles are written by a male lawyer, then the recommended articles will likely be of the same demographic attributes, discriminating against other groups. Therefore, to address this issue Oosterhuis and de Rijke (2020) created a state of the art policy aware counterfactual ltr approach which aims to take implicit feedback data such as clicks or time spent viewing an article such as data that captures each editors interactions with documents. This data is then integrated into the ml algorithm to produce an effective ltr model. This model has been proven to be effective, however, the Wikipedia dataset does not consist of such an amount of implicit data resulting individual fairness not being ideal in improving fairness in ltr.

The group fairness has a number of sub-categories with one of them being **demographic parity** which is an abstract approach of ranking documents. Demographic parity is defined as sensitive groups like gender and race which must have an equal distribution in the ranking. For example, when an editor searches for articles of a specific topic to edit, the ranked results must have an equal distribution of Males, Females or Non-binary in the ranking. This type of fairness has been used by Kamishima et al. (2012) to create a Prejudice Remover Regularize (PRR) which probabilisticly decreases prejudice against groups with lower distributions to provide results more equally. This technique has been proven to be effective in improving fairness at minimal cost of relevance. However, the PRR cannot be applied to multivariate dataset whose domain is large as for example the numerous types of occupations. So we are proposing a similar approach to the

PRR that can be adopted using a multivariate domain which will in turn will provide a fairer exposure of protected groups by providing boosts to this features that allow them to have a more advantagous exposure. In this dissertation we will refer to this approach as feature boosting.

Another crucial sub-category of group fairness that has been widely used is **equal opportunity**. Equal opportunity aims for similar documents/items to be placed in similar ranked order regardless of their demographic group. For example, if a user searches for an article in Wikipedia, then the returned documents should be relevant to all users, without favoring certain characteristics or users, but rather have a diversity of returned documents in the ranked list. This approach has been adapted by Hardt et al. (2016) using a post-hoc correction. The post-hoc correction method creates a good predictor model, meaning that it can effectively identify the relationships between variables to make accurate predictions. This model could be unfair, which the correction method corrects in order to make it fairer by taking into account a fairness feature such as race. However, Woodworth et al. (2017) has proven that such approach can fail for certain types of losses, even if the optimal predictor with respect to those losses is learned in the first step.

Furthermore, other researches such as Singh and Joachims (2018) have proposed an optimal probabilistic ranking to equalize exposure among groups, which is proven to improve the exposure of protected groups within the ranking, but does not necessarily improve the exposure of the top ranked documents. Having fair exposure in the top-n documents is important since users/editors just tend to only focus on the top-n documents in the list. There have been a number of studies such as Joachims et al. (2007) that show that users interact only with the top-n items from a vertical list of documents, with n be limited to the first page of the results. This has been characterised as the position bias, which is defined by Collins et al. (2018), as users often tend to pay more attention to documents placed at the top of the list even if those documents are not relevant. The limitations of post-hoc correction identify the importance of not incorporating fairness in the pre-processing part of learning. Thus, using this problem raised from post-hoc techniques we have identified that one of the main concerns of equal opportunity is that it might not apply to the top-n documents. Therefore our proposed approaches aim to use equal opportunity with some some demographic parity to not only improve fairness in the ranked list but target equal exposure in the top-n documents to prevent position bias.

2.2.1 Fairness in Learning To Rank

Fairness can be implemented in either of the 3-stages of ML processing. However, each stage has different types of strategies implemented. These stages can be seen in an illustrative example on **Figure 2.1** showing where fairness strategies can be implemented in the ml process.

The first stage is the **pre-processing**, refers to the stage before training a model, where appropriate data manipulation and cleaning occurs to remove noisy and unwanted data. This stage is an important in the ML process, however as it is discussed above, even though you could mitigate some fairness in this stage it is very limited to specific scenarios. Nonetheless, we will be using the pre-processing phase to retrieve important scores that will be used as fairness scores in the ltr process.

The second stage is the **in-processing** stage which is also known as training-time. An example of how to improve fairness at this stage is for models' training objective to be modified in order to incorporate fairness by adding constraints or regularization penalties. Such an approach was used by Agarwal et al. (2018) to create two reduction approaches for a sequence of cost-sensitive classification problems with a good relevance and fairness results. However, such methods require a lot of resources, when training the model, thus being computationally expensive meaning that it is not ideal for a non-profit organisation such as Wikipedia, as they only have a limited budget (Wikimedia Foundation n.d.). Therefore, in our dissertation we aim to provide a real life solution for Wikipedia using this limited budget and the resources available to the public.

The third stage is the **post-processing** phase, where fairness strategies are integrated into the model as feature scores and added to the ltr process as tweek values to re-rank the ranked list. These scores depend on the definition of fairness and what these fairness scores represent. Post processing methods learn transformations based on certain fairness features of model scores to achieve these fairness definitions. This is done by adding certain fairness scores into a feature list which is then used by the model to re-rank the ranked list based on the new feature scores. This approach was adopted by Nandy et al. (2020), where numerous group fairness techniques were implemented to re-rank a large dataset to make it fairer. This approach is the most suitable approach of solving fairness since we aim to apply demographic parity and equal opportunity without costing relevance. Thus, this is not possible in the pre-processing phase as relevance will drastically be degrade if we remove documents to achieve equal opportunity and in-processing is to expensive.

In addition, by using a post-processing approach we can easily implement a number of strategies which we then can compare.

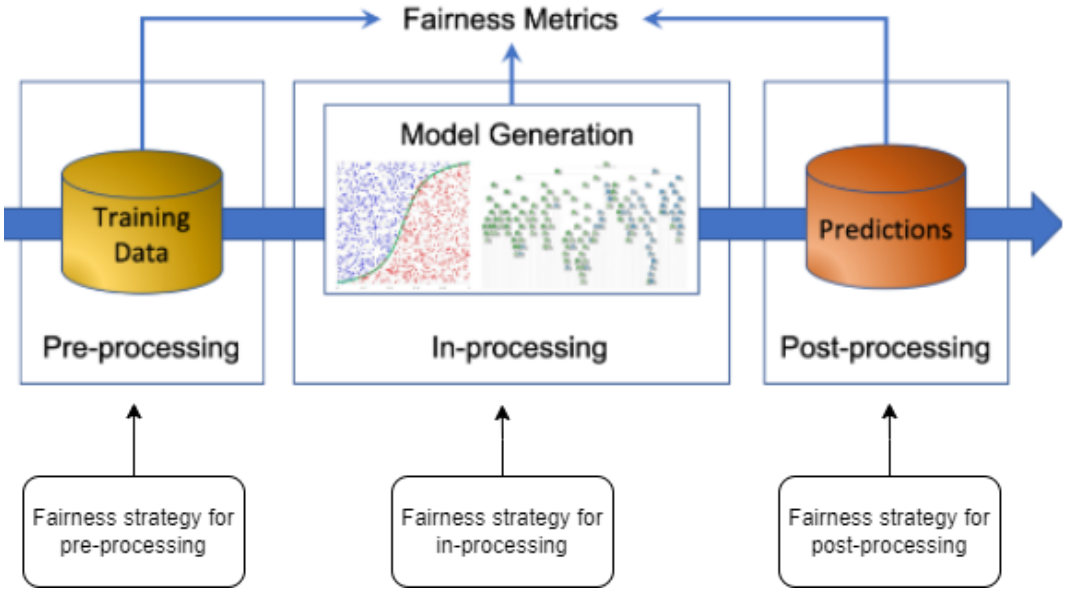


Figure 2.1: *This is a high level illustration of the pre, in and post processing of a machine learning process, indicating that fairness strategies can be applied to either, a combination or all of these stages. This diagram is inspired by Caton and Haas (2020)*

2.3 Fairness Features

As previously discussed fairness features are certain scores that will be used in the ltr process as re-ranking values that will induce fairness. Each fairness feature represents a certain attribute like gender of the author, the date of creation of the article and so on. So when selecting these re-ranking features a certain type of bias may be introduced into the model called biased feature selection. Biased feature selection is the process of selecting fairness features that are highly influenced by certain groups in the features. For example, if only the gender and race of Wikipedia authors are considered, of which both are dominated by males and white people respectively, it meaning that the resulting ranking will be from documents that are predominantly by white male authors thus, generating an unfair ranking towards other groups such as females. Conclusively, feature selection can be used as a fairness mechanism whereby by selecting specific features one can improve fairness.

To solve this problem, there has been an interest in developing techniques that will allow ir engineers to select features that will prevent biases in their models. An example of such technique made by Grgić-Hlača et al. (2018) is to use logistic regression and find the subset of the dataset that will maximize fairness at the lowest cost of relevance. Such technique has been proven effective when specific conditions are optimized; suggesting that to receive the most effective results appropriate tunning of the model is required. Another approach made by Dorleon et al. (2022) was to divide the features into two categories of redundant and non-redundant features lists. Then using these two lists multiple models are trained with different partitions of these list to find the most optimal combinations of features, with an enhanced performance in both accuracy and fairness. However, such ways of selecting features can be considered computationally exhausting when having big datasets. Thus, we propose an abstract method of deciding feature selection through data observation and distribution analysis using graphs to determine relationships and which groups can be unfairly treated by the ltr. As we have previously seen by **Figure 1.2** the number of active editors is highly unevenly distributed throughout the world, thus indicating that other characteristics of these articles can also be unevenly distributed.

2.4 Machine Learning Algorithm

In order to create a ltr model we also have to discuss about the numerous ml algorithms that can be integrated into ltr. One of the most highly popular machine learning algorithm is the Gradient Boosting Decision Tree (GBDT) which is renowned for its efficiency, accuracy and interoperability. It has been recognized for achieving state-of-the-art performance in a wide range of machine learning such as ltr as stated by Friedman (2001) however, it has also been found that there are big trade-off between accuracy and efficiency therefore making this type of algorithm unsuitable for the Wikipedia dataset. Alternatively, we are proposing of using a ml algorithm that is based on GBDT which is proven to be more efficient with larger datasets. This is the LightGBM algorithm which has been proven by Li et al. (2017) through comparing the lightGBM with a number of ml algorithms against numerous datasets. The reason behind this is due to the ability of lightGBM to use a number of given features and detect the relevance of a document based on these given features.This implies that this is a suitable algorithm for our strategies as we aim to introduce scores that will represent fairness.

2.5 Trec-Fair Ranking Track

As mentioned in **Chapter 1** the Trec-Fair aims to provide standardised tools for researches to develop and test new and innovative approaches on promoting fair exposure to a range of demographics or attributes represented by relevant documents in response to a search query.

Therefore it is important discuss the aims of these Tracks starting Trec-Fair 2020 Track Biega et al. (2019), which aimed to evaluate academic search systems in terms of providing fair exposure to different groups of authors, while also ensuring relevance to consumers.The central goal was to provide fair ranking results while balancing relevance and fairness, which is the abstract goal when implementing fairness in LTR. This is because by trying to implement fairer exposure of protected groups, you might remove relevant documents from the ranking, thus reducing relevance. The following years, the Trec-Fair 2021 Track Ekstrand et al. (2022), aimed to develop retrieval algorithms that provide fair exposure to demographics or attributes ranked by relevance in response to a search query. Such characteristics are topical content or authors. The end goal of these algorithms is to support Wikipedia editors in improving articles by providing them with a fairer exposure on these protected characteristics as under-represented groups can result in systematic biases. Thus highlighting the importance of the unequal representation of groups within the dataset, that result in systematic bias. It is apparent that systematic bias is not easily treated due to the fact the latest Trec-Fair 2022 **?** has a very similar aim and themed structure as

the Trec Fair 2021 implying that more research was required. Both Trec-Fairs' aim to improve the exposure of articles retrieved by editors to solve this bias however, their main difference is that the Trec-Fair 2022 focuses on, evaluating systems on how fairly they treat multiple protected characteristics and the impacts for particular subsets of those protected characteristics. As a result, we can observe the development of each Trec-Fair and how they examine various aspects of fairness in LTR. The most recent Track involves assessing multiple groups simultaneously, indicating an evolution in the approach to evaluating fairness. Therefore, given the lack of research in this area as at the time of this dissertation the where the Trec-Fair 2022 hasn't been completed, it has been selected as a base as it also involves in solving a real-world scenario which has a direct impact to society.

2.6 Metrics

When measuring the performance of a LTR model you should not only consider the relevance metrics but also the fairness metrics, since relevance metrics cannot measure fairness and vice-versa.

2.6.1 Relevance Metrics

Since, this dissertation is based on the Trec-Fair it is important to note what was already considered. In the Trec-Fair 2020 Biega et al. (2019) the relevance metric used was the **Expected Reciprocal Rank** (ERR) which takes into account, both the relevance of the documents and their position in the ranking. ERR has the advantageous properties of measuring not only the relevance of a document to a query but also make measurements according to it's position in the ranking. However, we are aiming to use the orinal form of ERR, the Reciprocal Rank (RR) due its wider use in the Informtion Retrieval world, that measures the relevance using the rank of the first relevant document.

Futhermore, there are other relevance metrics that have similar measurement capabilities and have a wider usage in the IR world. For example, Trec-Fairs 2021 Ekstrand et al. (2022) and 2022 **?** both use normalized Discounted Cumulative Gain (nDCG) to measure relevance, as it also uses the relevance of a document to a query and it's position. This type of metric is advantage as it measures the gain (change of position) of each item in the ranking. Thus, nDCG will provide us with more information about the general operation of the ranking.

Moreover, since different relevance metrics have different measuring capabilities we have also considered another metric that is widely used by researchers, the Mean Average Precision (MAP), as noted by McFee and Lanckriet (2010) which measures the average precision over a set of queries. However, MAP functions very poorly with imbalanced data and as our dataset consists of some highly imbalanced features.

Therefore as stated by Al-Maskari et al. (2008) the Number of Relevant Documents (NRD) acts as the father metric for other measures like RR and nDCG which somehow incorporate this relevance number. Also, Al-Maskari et al. has found a correlation between NRD and system effectiveness showing that NRD is a metric that can successfully measure positive performance of systems. Making this an ideal metric for our experiment.

Therefore, taking into consideration what the previous Tracks used and the world standards, this dissertation will use the Reciprocal Rank (RR), nDCG and Interpolated Precision-Recall (IPR) to measure Relevance. More about these metrics can be found in **Chapter 4.6.1**.

2.6.2 Fairness Metrics

Choosing fairness metrics is more complicated than choosing relevance metrics as it depends on how someone defines fairness. For example, in the Trec-Fair of 2020 Biega et al. (2019) a metric **disparsity** was used, aims to measure the exposure of each type of protected group in the ranking. However, since we will be using a number of protected groups to implement strategies, other metrics such as the **Attention Weighted Rank Fairness** (AWRF) metric is more suitable. This type of metric has been used in the 2 most recent Trecs and it is was developed by Sapiezynski et al. (2019), to measure the fairness of a set of ranked lists, using the population estimate and sample estimate(ranked list). This type of metric measures the attention of each group in the ranked list according to its position in the list thus providing valuable information of demographic parity and equal exposure within the ranking.

Furthermore, as we discussed above different types of fairness metrics can capture different information, thus by exploring even further into other types metrics such as Skewness. As far as we are concerned we haven't identified any research that uses skewness to measure fairness in ltr. Nonetheless, Rai (2020) has explained that skewness metric measures fairness by analyzing the distribution of a ranking across different groups of an attribute. For example, if some documents/items from one group has a higher exposure than another group in the ranking, then it indicates that the ranked list is unfair towards the group with the lower exposure. This kind of metric will allow us to indicate the amount of equal fairness in the ranking, where the lower values of skewness, the higher the equal exposure of protected groups.

2.7 Dataset

Another important characteristic in LTR is the dataset to be used. For example, Wikipedia provides access to people to look at statistics about the Wikipedia database. However, Trec-Fair has already collected this information and compiled the together to create the Trec-Fair 2021 and the Trec-Fair 2022. The Trec-Fair 2021 dataset by Ekstrand et al. (2022) which consists of Wikipedia articles. Each of this article has demoraphic features which can be used to improve fairness, such as the quality of the article and the continents that are associated with the article topic. This is a suitable dataset to use the previously mentioned Group Fairness approaches, but the Trec-fair 2022 has a similar but an expanded dataset(FAIR 2022). It has up to 22 different fairness features, thus providing a wider range of feature and technique combinations.These extra amount of features will allow us to implement several feature selection technique to improve fairness based on the relationship between these features. Thus highlighting the importance of having such a diversity of attributes that can be used in our proposed strategies.

2.8 Summary

Overall, this dissertation will be using the group fairness strategies during the post-processing stage of LTR as it is aims to fill the gap of how numerous re-ranking strategies can help reduce the unfair rankings in Wikipedia articles by ensuring that certain groups are not systematically disadvantaged or excluded by the ranking. This will be done by using techniques such as feature selection and feature boosting. Furthermore, the dataset to be used does not consist of implicit feedback and it consists of a number of attributes that through feature selection and applying demographic parity or equal opportunity techniques we can achieve greater fairness of these under-exposed groups at minimal cost of relevance.

3 Proposed Approaches and Implementations

This chapter aims to provide an insight into the strategies implemented. These strategies are divided into 4 section each section uses a different set of demographic information about the Wikipedia articles and focus on solving different aspects fairness for the articles. Each section will insist of 2 strategies, 1) based completely on feature selection , 2) A feature boosting strategy based on the previous feature selection. This chapter will reference a lot about the Trec-Fair 2022 dataset where more information about it and the features used can be found in **Section 4.2.1**.

3.1 Fairness distribution scores

Before explaining our proposed approaches it is vital to denote some definitions used by these designs. Firstly, our designs use information/attributes that describe geographical regions from all over the world. These regions are categorized into sub-continental regions, for instance the continent of Europe is divided into North, East, South and West Europe.

Secondly, we will use information chronological information/attributes that will appear as a range of dates or chronological age such as 2007-2011 and 20th-century respectively.

Thirdly, we use information/attributes that describe demographic information about the author of an article and demographic information about the article itself. These are gender, occupation of the author of the article and the page views, quality of the article. In addition, the term article author refers to the original creator of the article.

Fourthly, our proposed designs use pre-computed scores which are based on the distribution of each group of these attributes in the dataset by which we call distributional fairness scores. For example, if 50% of all articles their original author was a male, then the distributional fairness score of males is 0.5 as shown by **Figure 3.1**.

Dataset

DOCID	Gender	Occupation
1	Male	Athlete
2	Female	Politician
3	Male	Teacher
4	Male	Athlete
5	Other	Politician

(a) The gender and occupation attributes of article authors within a dataset of 5 documents

Distribution Results

Gender	Occupation
Male (0.6)	Athlete (0.2)
Female (0.2)	Politician (0.4)
Other (0.2)	Teacher (0.4)

(b) The distributional fairness scores of each group within the 2 attributes (gender, occupation)

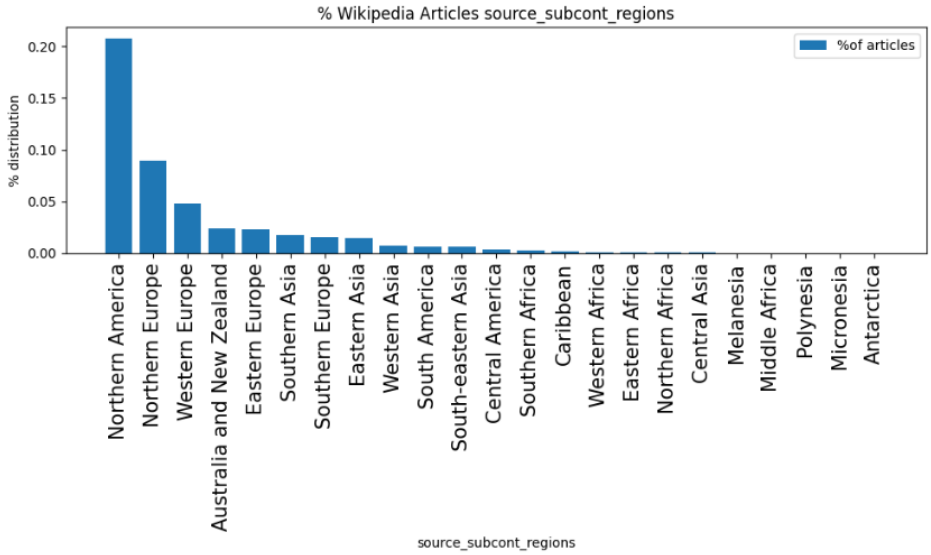
Figure 3.1: *(a)Shows the gender and occupation attributes of article authors within a dataset of 5 documents. The docid column represent the id of the documents. (b) Shows the distributional fairness scores of each group within the 2 attributes (gender, occupation). With the dominant group for gender being males and the dominant groups for occupation being politician and athlete*

3.2 Geographical Region Strategies

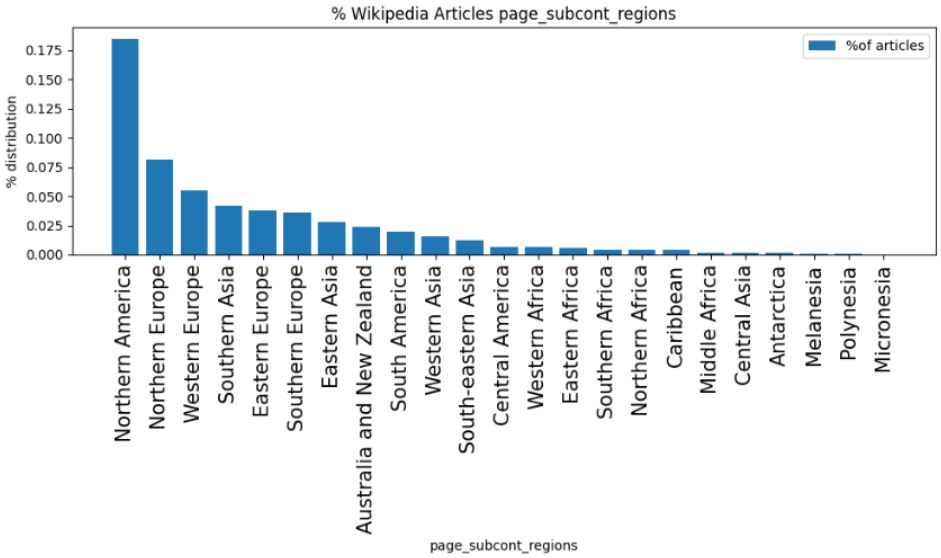
The following 2 strategies aim to use features that are associated with the geographic location on which the articles are based (articles source) and the geographic location associated with the article topic. These 2 attributes have exactly the same geographical regions.

3.2.1 Article Source/Topic Location (ASTL)

By analyzing the ranked distribution in **Figure 3.2**, it becomes evident that Northern America (NA), Northern Europe (NE), and Western Europe (WE) are the three dominant regions for both features. This implies that Wikipedia articles related to North America are likely to be edited by editors from North America, which can introduce bias towards certain topics as different regions might present events differently. Therefore, we aim to combine these two features and create a feature selection strategy that yields a higher relevance ranking while minimizing exposure to the lower distributional groups. However, as the remaining distributions are quite distinct, this approach may actually improve fairness. Regions with lower values in one feature are likely to have higher values in the other, resulting in a small boosting effect when these two features are applied in the ltr model. Consequently, a document that originates from Central Asia is highly likely to discuss North America, providing greater exposure to these less-represented regions.



(a) *The distribution of articles that were created in each region*



(b) *The distribution of articles that their topic is about each region*

Figure 3.2: *(a)Shows the distribution of article source between all sub-continental regions, with North America having the most articles and Antarctica having the lowest distribution . (b) Shows the distribution of the sub-continental regions for the article topic, with North America again having the biggest distribution and Micronesia having the smallest. The top 3 regions for both graphs are identical.*

3.2.2 Difference Article Source/Topic Location (DASTL)

This approach is founded on the feature selection strategy of the ASTL technique, as described in the **ASTL** section, which incorporates the origin location of an article and the geographical region it pertains to. By merging these two features, we can increase fairness in the lower ranked regions. Our objective is to enhance this fairness even further by boosting these regions. This will elevate their significance in the ltr model and provide them with a more equitable ranking. However, it is critical to identify the optimal value for boosting each region to strike a balance

between relevance and fairness. A high boost score can significantly impact relevance, whereas a low boost score may not improve fairness much. To identify the optimal score value, we examined the ranked distribution of regions in both features, as shown in **Figure 3.2**. We noticed that all regions beyond the top three have a dissimilar ranking in both features. Consequently, we calculated the absolute difference between the two features' distributional scores for each region to determine the optimal score value that can promote fairness with minimal effect on relevance. Suppose that, in the Wikipedia dataset, the source geolocation feature's distributional score for South Africa is 0.04, while the topic geolocation feature's score is 0.02. To boost the significance of South Africa in the ranking, we can compute the absolute difference between these two scores, which is 0.02. This value represents the amount by which we will increase the distributional score for both geolocation features to promote fairness throughout the dataset. This concept is further defined in **Equation 3.1**.

The 2 features used are identified as:

f_1 : The geographical region associated with the source of the article

f_2 : The geographical region associated with the topic of the article

$r \in \mathcal{R}$, where $\mathcal{R}$ is the set of all geographical locations of the world except (NA, NE and WE).

The DASTL score for a geolocation r is defined as:

$$DASTL_r \text{ score} = \left| f_{1_r} - f_{2_r} \right| \tag{3.1}$$

This new calculated scores will be added to the already known distribution score calculated by **equation 4.1**and the final score to be added to the ltr as a fairness feature score is defined like this:

$f \in \mathcal{F}$, where $\mathcal{F}$ is the set of fairness features used for the DASTL strategy.

$g \in \mathcal{F}_G$, where $\mathcal{F}_g$ is the set of groups within the fairness feature (geolocations), except (NA, NE, WE)

$$Fairness_{f_g} \text{ score} = distribution_{f_g} + DASTL_g \tag{3.2}$$

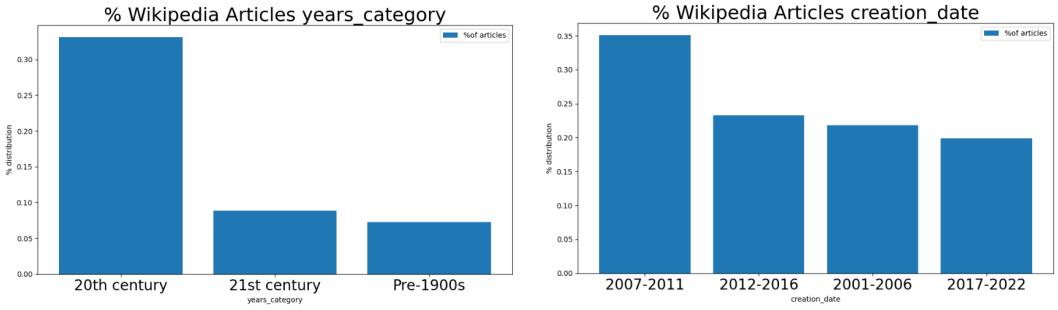
3.3 Chronological based Strategies

This section will describe 2 strategies that use the chronological dates at which the article topic talks about and the chronological date at which the article was created.

3.3.1 Article Creation date, Topic age (ACT)

As depicted in **Figure 3.3**, there is an uneven distribution of chronological dates associated with article topics, with the 20th century being the most popular and older articles having the highest exposure. This implies that editors are more likely to receive older articles on 20th-century topics between 2007-2011. Our objective is to enhance the exposure of groups with low exposure, such as older topics (Pre-1900s), with newer articles such as those from 2017-2022. To accomplish this, we combine these two features as fairness features in the ltr model. The creation date of the articles has a more uniform distribution across the chronological age of the topic of the article, which boosts the importance of the lower-ranked distributed features. For instance, if an article pertains to Pre-1900s, it will have a lower distributional score, but the next re-ranking feature will have a higher score since all groups of the article creation date have a high distributional

score, thereby elevating the lower-scored feature in the ranking. This approach should enhance both fairness and relevance since we are boosting lower-exposed groups that are also pertinent.



(a) *Distribution of articles, in each age of the topic.*

(b) *Distribution of articles, in each creation date.*

Figure 3.3: *(a) Shows the distribution of article between the topic age, with most articles being about the 20th century and the least being pre-1900s. (b) Shows the distribution of article creation dates between 4 date ranges with most articles being created in 2007-2011 and the least articles being created between 2017-2022*

3.3.2 Uniform Article Creation date, Topic Age (UACT)

The UACT strategy builds upon the feature selection approach of the previous ACT strategy, with the goal of improving exposure for under-represented features like the Pre-1900s chronological age topic. To achieve this, the UACT strategy implements feature boosting by assigning a positive boosting score to the two lower-ranked distributional features (21st century and Pre-1900s), while applying a negative boosting score to the highest-ranked feature (20th century), as illustrated in **Figure 3.3a**. The positive and negative boosting values were both set to 0.1, which strikes a balance between boosting disadvantaged groups and maintaining the importance of each group. Notably, even after boosting, the dominant group (20th century) remains the most important. To implement the UACT strategy, the computed boosting scores are added/subtracted from the distributional score of each article's topic group, as outlined in **Equation 3.3**.

The characteristics of the equation are:

$$f : \text{The chronological age associated with the topic of the article}$$

$$g \in \mathcal{G}, \text{ where } \mathcal{G} \text{ is the set of all ages that belong to f.}$$

$$x_g : \text{ is the distributional score according to g}$$

$$f(x_g, d) = \begin{cases} x_g - 0.1 & \text{if } d = \text{21st century} \\ x_g + 0.1 & \text{if } d = \text{20th century or Pre-1900s} \\ x_g & \text{otherwise} \end{cases} \tag{3.3}$$

In this strategy, we use the distributional scores of the creation date of the article, along with the boosted scores of the age associated with the topic of the article, as re-rankers for each document in the ranking. By decreasing the importance value of the major group and increasing the importance of the lower groups, we aim to achieve a more uniform distribution and improve the exposure of under-exposed groups in the ranking. However, it should be noted that while this approach may improve fairness, it may not significantly improve relevance.

3.4 Author & Article attributes

The following strategies will be using a number of information such as the gender of the article author, the occupation of the article author, the page views of the articles and the article quality.

3.4.1 Author Article Demographics (AAD)

Based on the highly uneven ranked distribution shown in **Figure 3.4** and **Figure 3.5**, it is evident that articles created by male authors, who are athletes, receive more attention from editors, while articles with lower page views are also likely to have lower quality. To address this issue and promote fairness, it is necessary to boost the exposure of lower quality articles with fewer page views, as these are more likely to be improved by editors. However, this approach alone may compromise relevance, as high-quality articles with high page views may be overshadowed. Therefore, we also consider author gender and occupation, as these features exhibit biases against certain groups, such as female computer scientists. By using these additional features, we strike a balance between fairness and relevance, thus improving both.

3.4.2 Boosting Imputing Author Article Demographics (BIAAD)

The BIAAD strategy is an extension of the AAD strategy, which takes into account author features such as gender and occupation, as well as article features like page views and quality. However, 70% of the author gender and occupation data is unknown, leading to lower distributional scores and reduced feature importance in the ltr model. As a result, groups like "Non-binary" and females are less likely to be ranked.

To overcome this issue, the BIAAD strategy aims to improve the distribution of gender and occupation features by predicting missing values and boosting underrepresented groups like females and "others". This approach balances relevance and fairness, with a greater emphasis on fairness.

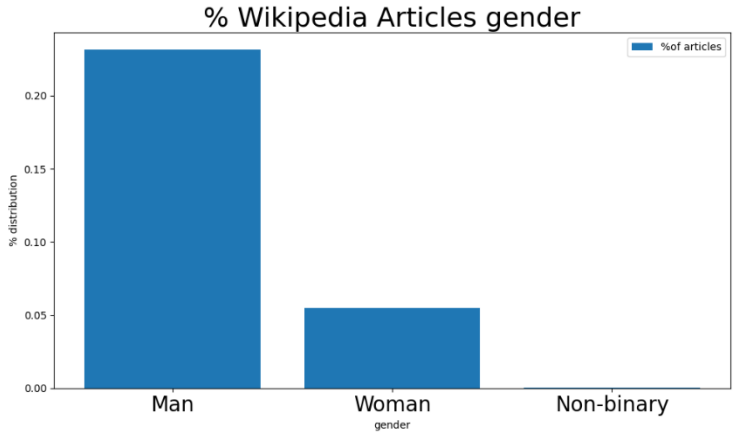
The BIAAD strategy uses four features, including gender and occupation, with approximately 70% of values missing. An imputation script is utilized to fill in missing values based on the known distribution of groups. The categorical values are converted into numerical labels using the LabelEncoder class from sklearn, and then the documents are divided into datasets with known and unknown values. The Decision Tree Regressor (DTR) is used to predict the unknown features based on the known features, and the predicted values are merged with the dataset of known values to restore it to its original form. The imputation process is illustrated in **Figure 3.6**.

After merging the imputed dataset, the BIAAD strategy employs a technique similar to the **UACT** strategy to uplift underrepresented groups. The major group (males) is negatively boosted by 0.2, while the female and Non-binary groups are positively boosted by 0.15 and 0.05, respectively. The specific boosting values are determined by the distributional proportionality of the groups to avoid compromising relevance or leaving only males in the top rankings. The mathematical formula for the boosting operation is clearly defined in **Equation 3.4**.

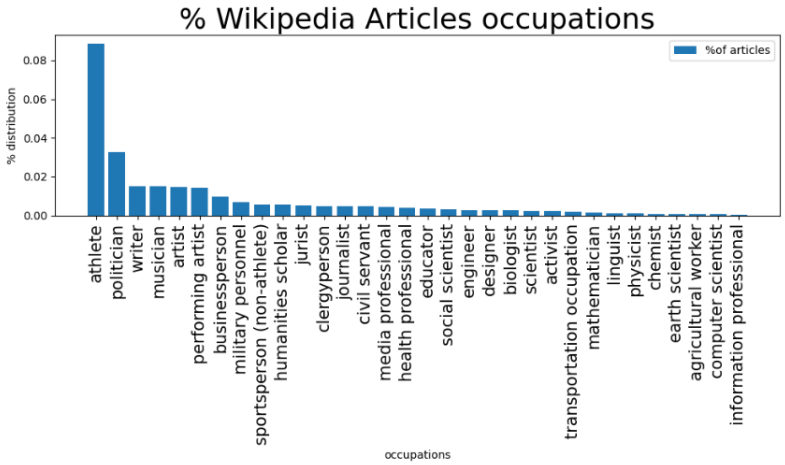
f : The gender of the author of the article

$g \in \mathcal{G}$, where $\mathcal{G}$ is the set of all genders taht belong to f.

x_g : is the distributional score according to g



(a) Distribution of articles, in each age of the topic.

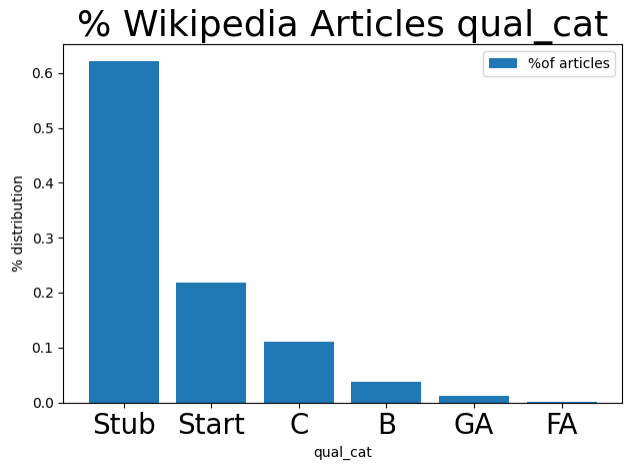


(b) Distribution of articles, in each creation date.

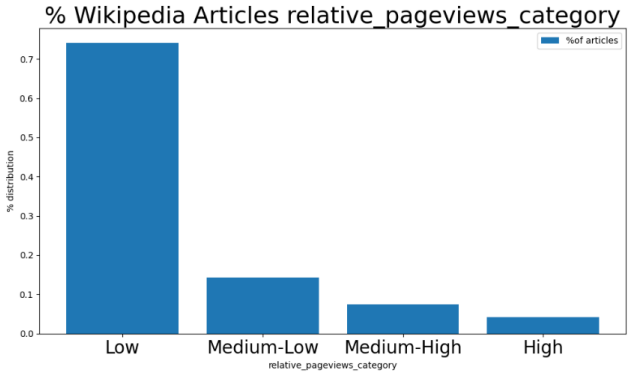
Figure 3.4: *(a) Show the distribution of genders of the original authors of articles. The dominant gender in Man and the least dominant is Non-binary (b) Shows the distribution of occupation of the original authors articles. The dominant occupation is athlete and the least dominant is Information professional.*

$$f(x_g, d) = \begin{cases} x_g - 0.2 & \text{if } d = \text{male} \\ x_g + 0.15 & \text{if } d = \text{female} \\ x_g + 0.05 & \text{if } d = \text{Non-binary} \\ x_g & \text{otherwise} \end{cases} \tag{3.4}$$

The BIAAD extends **AAD** strategy which consists of article author features such gender and occupation, but also article information such as page views and quality. Upon further invistagation about the dataset, we have identified that the majority (70%) of author occupations and gender are Unknown. Therefore, the distributional scores of gender and occupations are much lower, thus provinding a much smaller feature importance in the ltr. Therefore we aim to improve relevance and fairness by enhacing this 2 distributions of gender and female and then providing a boosting value that will boost the lower distributional genders such as females and Non-binary.



(a) Distribution of articles, in each age of the topic.



(b) Distribution of articles, in each creation date.

Figure 3.5: *(a) Show the distribution of article qualities, with the dominant group being the Stub (low) quality and the least dominant group FA (high) quality. (b) Shows the distribution of page views with the dominant group being the articles with low number of views and the least dominant group being the high number of views.*

3.5 All Feature Strategy

3.5.1 All Feature Included (AFI)

The AFI strategy addresses the broader issue of unfair exposures on Wikipedia, which is highlighted by the feature selection strategies ASTL, ACT, and AAD. These strategies reveal how geographical, chronological, and author/article features can unfairly expose lower ranked distributional groups. As illustrated in **Figure**, this means that editors are less likely to edit an article created by a female earth scientist from Southern Europe in 2022 with low-quality content and views about North America in the 20th century compared to an article created by a male politician from Northern Europe in 2009 with low-quality content (Stub) and views about the same topic.

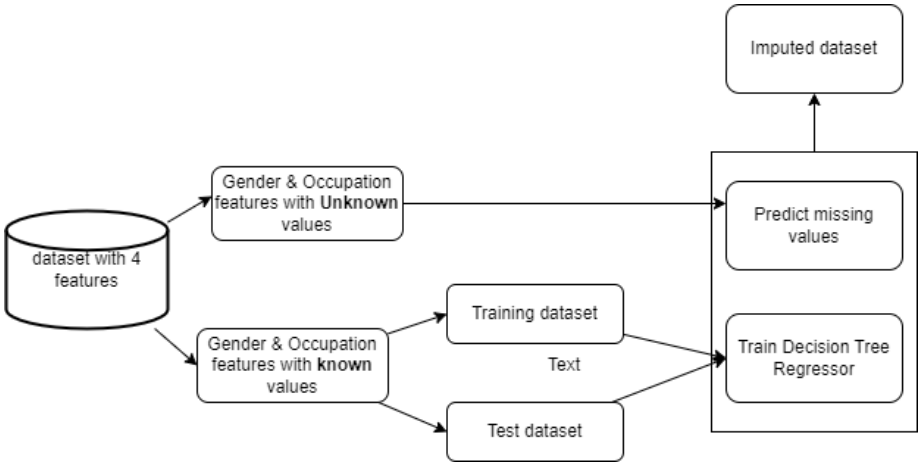


Figure 3.6: *The figure shows the process of creating an imputed dataset. This process begins by first divididing the dataset into documents with unknown and known values. Then we split the dataset of known values to training and testing data in order to train the Decision Tree Regresor(DTR). Then the DTR will use the dataset with the unknown data to predict this missing values. The result will be the imputed dataset.*

Table 3.1: *This table shows the sum of all distributional scores (approximate values from Figures 3.2, 3.3, 3.4, 3.5) of 2 documents that are made by 2 authors that have a different gender, occupation, location of where they created the article, age associated with article topic, but they have the same location associated with the article topic, the same category of article views and article quality. From this table we can see that Document 2 with a score of 2.63 has a higher distributional score as it is associated with attributes that make up of a bigger portion of the article distribution. The attributes in bold show the highest distributional values between the 2 documents.*

Article Attribute	**Document 1**	**Document 2**
Gender	Female (0.05)	**Male (0.24)**
Occupation	Earth Scientist (0.00)	**Politician (0.03)**
Article topic age	20th-century (0.34)	20th-century (0.34)
Article creation date	2017-2022 (0.20)	**2007-2011 (0.35)**
Location of article creation	South Europe (0.02)	**North Europe (0.09)**
Location of article topic	North America (0.18)	North America (0.18)
Article views	Low (0.7)	Low (0.7)
Article quality	Stub (0.6)	Stub (0.6)
Total Score	2.09	**2.63**

Nonetheless, the similarity in score between both documents is evident from **Table 3.1**, which is advantageous for the under-exposed documents in the ltr. This is because the feature importance in the ranking and the distribution of groups within each attribute are heavily imbalanced, leading to a more even feature scoring overall. As a result, a simple feature selection strategy utilizing all of the previously mentioned attributes is implemented to enhance both relevance and fairness.

4 Experimental Set Up

The following section describes the experimental set up used to implement and evaluate the designs proposed in **Chapter 3**, along with information about the dataset, models and metrics that were used.

4.1 Research Questions

Following the designed approaches mentioned in **Section 3**, we have come up with some specific hypothesis that we want to test.

- Which type of feature (e.g. Geolocation, Chronological, Article Author Attributes or all together) have the most significant impact on improving fairness for Wikipedia editors?
- What is the impact of incorporating LTR models on reducing bias and improving fairness in the ranking of the Wikipedia articles?
- Does feature selection or boosting of features have a the most positive impact?

4.2 Dataset

This section will provide detailed description of the Trec Fair 2022 dataset along with information of why it was chosen.

4.2.1 Trec Fair 2022 Dataset

The Trec Fair 2022 dataset was the perfect candidate compared to the previous Trec Fair datasets because it consists of up to 23 fairness characteristics that can be used to create and implement new strategies. Moreover, it has come to our attention that the initial statistics regarding the Trec-Fair 2022 have not been made available to the public, thus we don't have information about the number of articles used. Thus, referencing from the **Table 4.1**, which are the statistics of the indexed dataset, we cab deduce that there are enough documents and terms in each document to have sizeable dataset that can act as a population, and draw conclusions from this dataset.

Table 4.1: *Trec Fair 2022 index Collection Statistics*

Collection Statistics	
Number of documents	6 475 537
Number of terms	7 174 631
Number of postings	989 531 717
Number of tokens	1 972 209 736

Even though, this dataset consists of 23 fairness features, only 8 of them has been chosen out of the 23 because they can be summarised into categorical data and be able to detect biases based on groups within the fairness attribute. The fairness features used can be found on **Section 4.2.2**.

4.2.2 Fairness features

qual_cat Each article has a quality score from 0 to 1. These scores have been categorized into 5 categories Stub, Start, C, B, GA, FA from low to high.

source_subcont_region The subcontinental region from which the source of the article has originated from. There are several regions from all over the world, including Unkown and NaN for documents that their location was unavailable.

page_subcont_region The subcontinental region from which the topic of the article is about. There are several regions from all over the world, including Unknown and NaN for documents that their location was unavailable.

creation_date_category The date range between 2001 and 2022, that the article was created. The categorical ranges are 2001-2006, 2007-2011, 2012-2016 and 2017-2022.

years_category The year age category of the topic of the document. The are 4 categories are Unknown, Pre-1900s, 20th century, 21st century. The Unknown category represents documents that don't have a topic age or it is unavailable.

occupations The occupation of the author of each document. There is a big range of occupations and includes a category for Unknown for and NaN for documents that their occupation was unavailable.

gender The gender of the author of each document. There are 4 categories in the gender feature, Male, Female, Non-Binary and Unknown.

relative_pageviews_category The category of page views that each document has. The number of page views is split into 4 categories Low, Medium-Low, Medium-High, High categories.

As seen from the list above, some of the fairness features have categories like Unknown and NaN, that represent fairness categories that are not applicable or unavailable for specific documents. Therefore, when calculating the fairness scores for each feature, values that are unavailable, will be granted a score of 0. This is because such values shouldn't have an impact in the ltr model since they don't represent a protected group. Furthermore, some features have been observed to have a huge percentage of Unknown/Nan values as shown in the **Table 4.2**. Therefore if these values included in the training features then they would have a huge impact on the models, as they would have been trained on missing information.

4.2.3 Dataset Components

The Trec Fair 2022 dataset is made out of 3 important components:

The first component is a dataset of all the documents in the dataset where each document having a docid and 23 other columns of information as discussed and shown in **Section 4.2.2**.

The second component are the topics (queries) used to actually train the ltr models. There are exactly 50, topics which will be used to train the ltr models, of which 70% of them are randomly selected as train data and the other 30% as testing data. It is important to note that techniques such cross-validation and k-fold have been considered in order to maximize the performance of the models. However, such approaches are not beneficial for this dissertation as each approach will be using a different set of features, thus in order to maximise the performance of each approach, different training datasets might be used. Moreover, the aim is to find strategies that will overall improve fairness without having a negative impact on relevance, thus maximising performance is out of the score of this dissertation.

The third component are the qrels also known as ground truth. The ground truth is a table which matches all the queries to their relevant documents, so that we can use at the evaluation stage and test the effectiveness of the dataset.

Table 4.2: *Percentages of values that correspond to Unknown/NaN in each feature. From this table we can observe that not all features have 100% Known values.*

Unknown/Known distributions		
Fairness Feature	Unknown (%)	Known (%)
qual_cat	0	100
first_letter_category	0	100
occupations	73,64	26,36
source_subcont_regions	52,85	47,15
page_subcont_regions	42,58	57,42
gender	71,37	28,63
creation_date	0	100
years_category	50,74	49,26
relative_pageviews_category	0	100

4.2.4 Data scrubbing & feature scores

One of the reasons why the Trec Fair 2022 dataset was selected for this dissertation is due to its numerous fairness features. However, not all features can be used as some cannot be categorized. Therefore, the goal was to convert features such as article quality, which is represented as a float value between 0 and 1, into a categorical value indicating from low to high quality, and replace the float values with the categorical values. This was done for all of the attributes mentioned in **Section 4.2.2.**

After creating the new dataset of categorical values, we computed the global distributions of each group in each of the attribute/feature and then we divided it by the total number of documents to obtain a normalised feature scores. Normalizing the features is crucial for treating all features equally on the same scale during training, as emphasized in a study by Singh and Singh (2020). These normalized distribution scores serve as feature scores in the ltr models and provide a real-world description of the data. This normalised distibutional scores are the fairness scores that we mentioned in **Section 3** as distributional fairness scores. This scores can be better described by the **Equation 4.1**

$f \in \mathcal{F},$ where $\mathcal{F}$ is the set of fairness features such as gender of author, ...

$g \in \mathcal{F}_G,$ where $\mathcal{F}_g$ is the set of groups within the fairness feature f.

n is the total number of documents in the dataset.

$$distribution_{f_g} \text{ score} = \frac{1}{n} \sum_{g \in \mathcal{F}_g} g \tag{4.1}$$

4.3 PyTerrier

As noted by Craig Macdonald (2020), the advent of deep machine learning(ML) platforms such as Tensorflow and Pytorch in python has heightened the need for an information retrieval platform that enables users to conduct experiments in a user-friendly and meaningful manner. Therefore main library used is PyTerrier which covers all the features said. Additionally, the optimization of retrieval pipelines through automatic methods enhances their efficiency and adaptability to specific IR platform backends, leading to a more optimized user experience and efficient IR

process. This is due to the considerable big size dataset that we will be discussing below and thus the need for efficient IR platform. Furthermore, PyTerrier provides easy access to retrieval models such as BM25 and PL2 which are relevance scoring models used in ir. Also, PyTerrier has a simple integration's with Sklearn ML algorithms which allows us to easily employ our strategies.

4.3.1 Information Retrieval Models

Here we discuss all the Information Retrieval models used and how we used them to create our model.

BM25 estimates the relevance of a document to a given search query based on the frequency of query terms in the document and the length of the document.

TF estimates the relevance of a document to a given search query. It assumes that the relevance of a document is directly proportional to the frequency of the query terms in the document. In this model, the frequency of each term in the document is calculated and used to determine the document's relevance to the query.

PL2 estimates the relevance of a document to a given search query. It takes into account the length of the document, query term frequency, and term frequency saturation to provide a more accurate estimation of relevance.

FeatureBatchRetrieve is a retrieval model used in PyTerrier, which adopts machine learning techniques of ranking documents based on their relevance towards a given query. This model uses a set of pre-defined features, such as the scores calculated by BM25, TF or PL2, to re-rank the documents based on these features. These features are then also used by a ltr algorithm to learn the reranking process using these features.

4.3.2 Information Retrieval Process used

Now that we have defined these ir concepts we will proceed with the describing the process used. Firstly, we have a created pipeline model which where our ltr approaches will be applied to. This pipeline is a FeatureBatchRetreieve model that will firstly remove all irrelevant documents from the ranking and then using the BM25 model we retrieve the most relevance documents and rank them. Once, this operation is completed then these documents will be re-ranked based on **Term Frequency** and **PL2** which will act as features scores in a feature list that each document in the ranking will have. We are using a BM25 model as our first ranker and then we re-rank these documents using Term Frequency and PL2 models. The definitions of the models mentioned can be found in **Section 4.3.1** and the process described can be better seen by **Figure 1.1**

4.4 Learning To Rank Set Up

To talk about the ltr set up we first have to discuss the ml algorithm used and crucial decision and paramters about it.

4.4.1 Models

As noted in **Section 2.4** different machine learning models, will produce different relevance and fairness scores for the same dataset, as each model is trained using different techniques. Some models will have a better performance in fairness while other won't. Therefore, we introduced the LightGBM which has been proven to work computationally much better in ltr approaches as denoted by Li et al. (2017).

Light Gradient Boosting Machine (LightGBM) is main ml algorithm used, which is a tree-based algorithm using a gradient boost framework. The main benefits of this model are that it is designed in such a way that it is highly scalable and can handle large datasets with numerous features resulting in a faster and memory efficient training process. This is a perfect fit for this experiment the dataset used contains up to 6.4 million documents. Furthermore, LightGBM is used by all baseline and strategies algorithms through out the experiment with it's default parameter settings as shown in the Documentation (2021). We use default parameters in order to identify the best strategy for the Wikipedia problem, without incurring the costs associated with hyperparameter tuning on large datasets such as Trec-Fair 2022, as highlighted by Snoek et al. (2012). As we have previously discussed, Wikipedia is a non-profit organization with a limited budget.

Moreover, it is essential to discuss about the importance of feature importance. It is a measure of the relative contribution of each input feature to the output of a ml model. It helps to identify which features have the most significant impact on the model's predictions and can be useful for understanding the model's behavior, improving its performance, and identifying potential bias or unfairness. This type of measure will help full to identify which features are important to the Trec-Fair 2022.

4.4.2 Integrating features in LTR

In order to talk about our ltr implementations it is important to first talk about how ltr works with PyTerrier. It uses a set of features to represent each query-document pair, such as scores computed by the FeatureBatchRetrieve and fairness scores. These features are then used to train a machine learning model (LightGBM) as previously mentioned.

In order to incorporate our proposed designs into the LTR we used PyTerriers custom Transformers classes. A transformer is PyTerrier is a class that takes as input a dataframe and using a *transform*() method, several custom operations can be implemented to output a modified dataframe. Each strategy is implemented using one of these custom classes as shown in **figure 4.1**.

```
class MyScorer(pt.Transformer):

    def transform(self, input):

        # Calls a function that computes fairness scores
        # according to the strategy employed
        fairness_scores = compute_fairness_scores()

        # Add scores to the inputed ranked dataframe
        ...
        return input

    def compute_fairness_scores():
        ....
        return scores
```

Figure 4.1: *MyScorer is a PyTerrier transformer class that uses a computed_fairness_scores() function to create fairness scores based on the strategy used, which is then integrate with the column features from input dataframe and then returned*

Each Scorer class follows a similar transformation method, since the way of retrieving and adding scores is almost identical for all strategies.

Therefore, in order to create our strategies we have have applied our transformer classes into the pipeline which is then applied to the LTR model. This, will allows us to add our fairness scores into the features list created by the FeatureBatchRetrieve which will be used by the ml algorithm. All of our strategies have a similar format to **figure 4.2** when training the LTR model.

```
lgbm = lgb.LGBMClassifier()
lgbm_pipe = pipeline >> MyScorer() >> ↪
        pt.ltr.apply_learned_model(lgbm)
lgbm_pipe.fit(train_topics, qrels)
```

Figure 4.2: *The figure shows how the FeatureBatchRetrieve is then applied through the custom transformer class called MyScorer, which then is used as an input into the lightgbm ltr algorithm. Lastly this model is trained based on a set of trained data and qrels*

Thus, each approach will create it's own number and values of features that will allow the ltr model to learn more complex relationships within the dataset. Moreover, for each topic we are only retrieving the top 100 documents because as we have previously mentioned there is a position bias that editors will not go through all of the ranked results. The provided example below in **Figure 4.3** shows the output of the ltr model of ranked list in ascending order from top to bottom for the query 'agriculture'.

Figure 4.3: *The table shows how a ranked list is based on a query "agriculture" which has as id 1. The docid column shows the document id of each document in the ranking with the rank column showing the ranking of these documents with 0 being the most relevant (top ranked) document and 99 being the least relevant out to the top-100 documents. Each document consists of a features list with scores that are used by the ltr to learn rankings. The first numbers of the list are the relevance scores which in this case are scores created by the tf and pl2 ir models respectively.*

Output example				
qid	query	docid	features	rank
1	agriculture	515648	[81, 6.3745, ...]	0
1	agriculture	5416574	[69, 6.5915, ...]	1
1	agriculture	151484	[65, 6.1755, ...]	2
...	...	...	...	...
...	...	...	...	...
1	agriculture	2351446	[18,6.3015, ...]	97
1	agriculture	58	[10, 6.2157, ...]	98
1	agriculture	844656	[11, 6.1245, ...]	99

4.5 System Set Up

This experiment was done on Virtual Machine with a linux based Operating System. This is because at the time of the experinment PyTerrier was only compatible with a Linux based operating system. Moreover, the experinment requires a minimum of 12 Giga Bytes of RAM available due to the huge size of the dataset and a minimum of 40 Giga Bytes of hard disk memory with a recommended amount of 4 processors. Futhermore, several python libriaries are being used such as PyTerrier, SkLearn and matplotlib.

4.6 Evaluation Metrics

The following section we will firstly outline the Relevance Metrics used to measure how relevant are the ranked documents to each topic. Then, we discuss the Fairness Metrics used, to measure the fairness score of the ranked documents and how fair the ltr is according to the fairness features. Lastly, we discuss the type of statistical analysis done to measure statistical significance.

4.6.1 Relevance Metrics

There are numerous ways of measuring Relevance in IR models, however this dissertation focuses on 3 of them, reciprocal rank (RR), Number of Relevant Documents retrieved (NRD) and the Normalised Discounted Cumulative Gain (nDCG) as discussed **Section 2.6.1**.

Reciprocal Rank (RR) is a relevance metric that measures the effectiveness of a retrieval model based on the rank of the first relevant document. Higher values of RR indicate a more effective systems which means that the first relevant document is higher in the ranking as shown by the **Equation 4.2**.

$$RR = \frac{1}{K} \tag{4.2}$$

where K, is the position of the first relevant document in the ranking

Number of Relevant Documents (NRD) is a relevance metric that measures the number of relevant documents in the ranking. In our case we will be using 15 topics as our test_data which is 30% of the topics as discussed in **Section 4.2.2**. Therefore, each model will consist of 1500 ranked documents as shown in the **Equation 4.3**

$$15 \text{ topics} \times 100 \text{ No. of documents per topic} = 1500 \text{ ranked documents} \tag{4.3}$$

The NRD will measure the number of relevant documents out of these 1500 documents, where a higher number indicates a better performance as more relevant documents are retrieved.

Normalised Discounted Cumulative Gain(10) (nDCG(10)) measures relevance by calculating the sum of these relevance scores of the top 10 documents retrieved by the ltr model. Then these scores are discounted according to their position in the ranking and normalised. A higher value of nDCG at 10 means that the model has effectively returning the most relevant results to the user's query within the top 10 search result as it is shown in the **Equation 4.4**

$$\text{nDCG@10} = \frac{\text{DCG@10}}{\text{IDCG@10}} \tag{4.4}$$

where k is the position cutoff for the ranking list, which in our case is 10. DCG is the Discounted Cumulative Gain, and IDCG is the Ideal Discounted Cumulative Gain.

The Discounted Cumulative Gain (DCG) is calculated as:

$$\text{DCG@k} = \text{rel1} + \sum i = 2^k \frac{\text{rel}_i}{\log_2(i+1)} \tag{4.5}$$

where rel_i is the relevance score of the item at position i in the ranking list.

The Ideal Discounted Cumulative Gain (IDCG) is calculated by taking the DCG of the ideal ranking list, which is a ranking list where the items are sorted in descending order of their relevance score. The formula for IDCG is the same as DCG, but the relevance scores are based on the ideal ranking list.

4.6.2 Fairness Metrics

In this dissertation, 2 types of fairness metrics are used to measure the fairness scores of the fairness attributes of the resulted ranked documents, Mean Attention Weighted Ranked Fairness (mean AWRF) and skewness using the Pearson coefficient.

Mean Attention Weighted Ranked Fairness is a measured made by Sapiezynski et al. (2019). The algorithm measures fairness by calculating the distance between the exposure distribution and the estimated population distribution, with smaller distances indicating potentially fair distributions. A mean distance for a specific lambda value. Moreover, as we will be evaluating models on more than 1 fairness feature, we have taken this algorithm a step further and evaluate fairness based on the mean of all the awrf feature scores. This will allows to make a comparison betweeen each mean-awrf measure. Also, our fairness metric has been inspired by Sapiezynski et al. (2020) where a lower mean-awrf indicates a fairer ranking.

Skewness measures fairness by evaluating the degree of asymmetry in the distributions of data. For example, in the feature gender if most of the data are males then there the skewness value will be high. The aim of skewness in ltr is to have a value closer to 0 which means that the data are symmetrical.

4.7 Evaluation Set Up

This section will provide information on the type of significance testing performed along with other types of evaluation done to accepted or deny the research questions stated in **Section 4.1**

4.7.1 Baseline Models

For this experinment we have designed 2 seperated baselines models that each will examine a different aspect of the experinment.

The first baseline model is a simple BM25 ir model which has been described in **Section 4.3.1**. By using the BM25 as a baseline we try to identify if any of the strategies implemented have performed better than the BM25.

The second baseline model is LightGBM ltr model with the only features scores being those of TF and PL2. This type of baseline aims to identify if the strategies implemented have an improved significance in both relevance and fairness.

4.7.2 Significance Testing

In order to prove that our models have a better relevance and fairness score we have to implement a Significance Testing between each strategy and the baseline models for each metric used. The t-test chosen for this experiment is paired t-test which is used to compare the means of two related samples, where each observation in one sample is paired or matched with a corresponding observation in the other sample as described byPeyton Jones (2017). Furthermore, in order to make these tests we had to make some assumptions:

- 1. The 2 models are used are independent from each other
- 2. The 2 models have the same number of observations

To generate a t-test for the relevance metrics we used PyTerriers in-built method of creating pair t-test during the experiment, where as for the fairness metrics we had to collect all the means from each of the 15 topics (15 fairness scores per metric) and create t-test using the scipy python library to compute a paired t-test using the ttest_rel() function.

4.7.3 Feature Importance Analysis

Feature importance can provide insights into the relative importance of each feature in the model's predictions along with information about the relationship between features. In feature analysis it also important use other methods of analysing features such as computing the relative importance scores of features with strategies that have common features such the Chronological strategies in **Section 3.3.2**. This will inform us if the boosting strategies implemented were effective in increasing the importance of features.

5 Evaluation

This chapter will present the results of our experiment, which aimed to compare the performance of our design strategies against a set of 2 baseline models. We will further discuss our finding and provide insights into the strengths and weaknesses of each of our strategies, as well as the implications of our results for the research questions and objectives of our study.

5.1 Results

This section will present the results gathered during the experiment, which can be seen on **Table 5.1**. From this table we can see that the best performing strategy was the ASTL with achieving the best scores for NRD, nDCG@10 and awrf with achieving the highest RR from all the strategies implemented. However, the model with the best RR was the baseline model of BM25, with the model that had the best skewed performance being the DASTL strategy. The strategy with the worst performance in RR was the AFI which also had the worst performance in nDCG@10. The ACT and UACT strategy had the worst NRD values with ACT having also the worst skewness valus. Overall, none of the results have shown a significant p-value in both of the baseline models.

Figure 5.1: *This table shows the experimental results of all the models used, with the LightGBM and BM25 acting as the baseline models. From this table we can see that the BM25 has the highest Reciprocal Rank values with the AFI strategy having the lowest RR. The ASTL has shown a good performance in both relevance and fairness as it has achieved the best results in NRD, nDCG@10 and awrf with DASTL having the best skewness score. ADD and BIAAD having the lowest NRD and nDCG@10 respectively. The AAD also had the worst awrf score and the ACT strategy having the worst skewness score. The has been no significance values, indicating that none of the models have made an improvement in relevance. Each arrow besides the metrics names shows the direction of effectiveness (↑ higher values are better), (↓ means better). In addition statistical significance for with baseline model the BM25 is denoted as (#) and statistical significance with baseline model the LightGBM is denoted as ().*

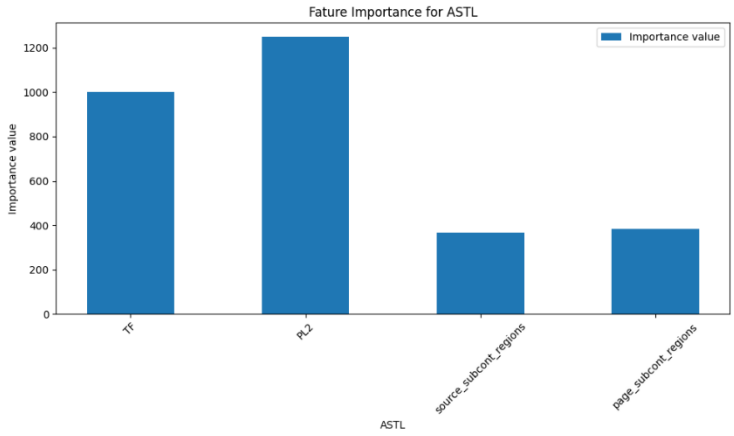
Results					
Model names	recip rank ↑	NRD ↑	nDCG@10 ↑	awrf ↓	skewness ↓
BM25 (base)	**0.7651**	714	0.4998	0.0769	2.5172
LightGBM (base)	0.5996	720	0.4899	0.0525	2.4174
ASTL	0.7333	**731**	**0.5044**	**0.0244**	2.1932
DASTL	0.6530	729	0.5020	0.0304	**2.1312**
ACT	0.6808	707	0.4895	0.0281	2.7716
UACT	0.6808	707	0.4900	0.0281	2.7715
AAD	0.5875	698	0.4521	0.0921	2.3995
BIAAD	0.5754	718	0.4487	0.0842	2.3726
AFI	0.5648	729	0.4385	0.0470	2.3256

5.2 Feature Importance

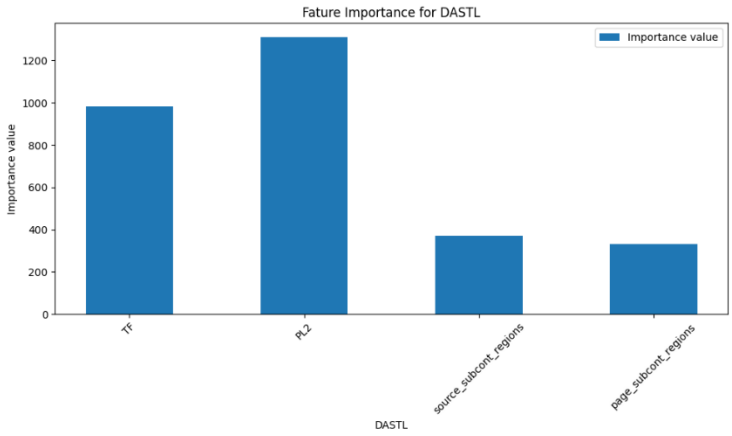
This section will provide a feature analysis for the proposed strategies.

5.2.1 Geolocation Features

By analyzing the features, we can determine that Pl2 is the most important feature for both strategies, while the geolocation associated with the source of the article and the geolocation linked with the topic of the article have the least impact on ASTL and DASTL, respectively. This suggests that the strategy proposed in Section 3.2.2, aimed at promoting under-distributional regions, has actually reduced the importance of fairness features in the ltr, thereby undermining their impact. Furthermore, **Table 5.1** shows that while boosting lower distributional locations helped to slightly decrease the skewness of the distribution, it also had a negative impact on relevance and fairness throughout the ranking. Thus, it appears that the boosting strategy only disadvantaged geolocation fairness features instead of benefiting from them.



(a) *Feature Importance for ASTL strategy*

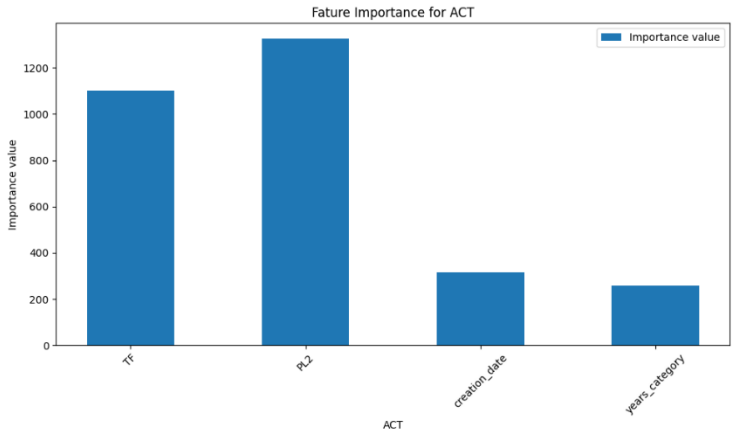


(b) *Distribution of articles, in each creation date.*

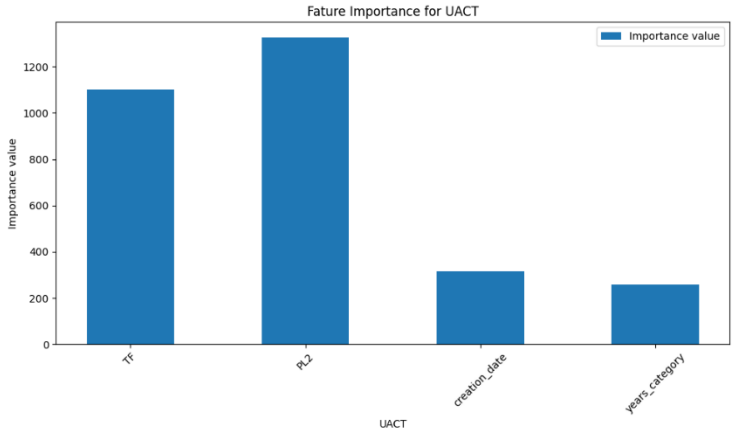
Figure 5.2: *(a) Shows the feature importances of the ASTL strategy for each feature with the PL2 being the highest and the source_subcont_regions (Geolocation associated with the source of the article) being the lowest. (b) Shows the feature importances of the ASTL strategy for each feature with the PL2 being the highest and the page_subcont_regions (Geolocation associated with the topic of the article) being the lowest.*

5.2.2 Chronological Features

By using the **Table 5.3** to assess the chronological strategies reported in **Section 3.3.1** the ACT and UACT, it was identified as the PL2 feature was the most important, while the year_category (which represents the chronological age associated with the topic) was considered the least important. In addition, it can be observed from **Table 5.1** that the boosting strategy had little to no effect on increasing the importance of the year_category feature. This conclusion is supported by the fact that the ACT and UACT strategies exhibit identical results in terms of RR, NRD, and awrf, while displaying comparable results in nDCG and skewness. Furthermore, the UACT strategy appears to slightly perform better.



(a) Feature Importance for the ACT strategy



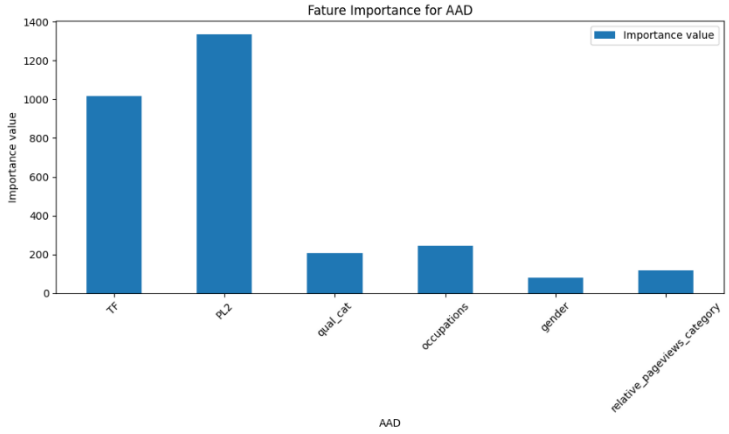
(b) The feature importance for UACT strategy

Figure 5.3: *Both (a) and (b) show the feature importances of the ACT and UACT strategies respectively with both having, PL2 as the most important feature and year_category (chronological age associated with the topic) the least important feature.*

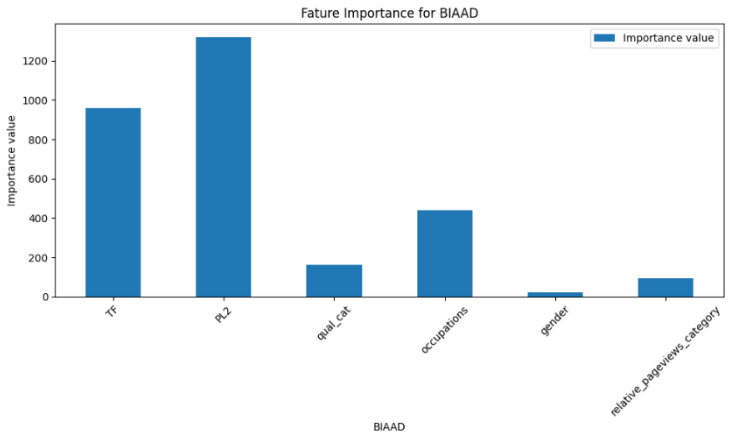
5.2.3 Article Author Features

From the **Figure 5.3** we can see that both AAD and BIAAD show the feature importances of the ACT and UACT strategies respectively with both having, PL2 as the most important feature and the gender of the article author is the least important feature. Also, with referencing to **Section 3.4.2** we can see that the boosting and imputation approaches have decreased the importance

of the gender feature, however they have greatly increased the importance of the occupation feature. The fact that the occupation feature received a higher importance it means tht it was able to influence the ltr model and greatly increased the amount of NRD of the BIAAD compared to the AAD. But nonetheless, reduced the relevance of the top-n documents. This is due to the much higher importance given to the occupation feature where there will be a greater exposure of occupations in the ranking, however not many of them will be important enough to be placed at the top-n of the ranking.



(a) Feature Importance for the AAD strategy



(b) The feature importance for BIAAD strategy

Figure 5.4: *Both (a) and (b) show the feature importances of the AAD and BIAAD strategies respectively with both having, PL2 as the most important feature and the gender of the article author is the least important feature.*

5.2.4 All Fairness Features

The **Figure 5.5** shows the feature importances of the AFI strategy which is described in **Section 3.5.1**. The most important feature is the PL2 with he least important feature being the gender of the author. Overall, the AFI strategy has better fairness scores from the baseline models but much lower relevance fromt them as shown by the **Table 5.1**

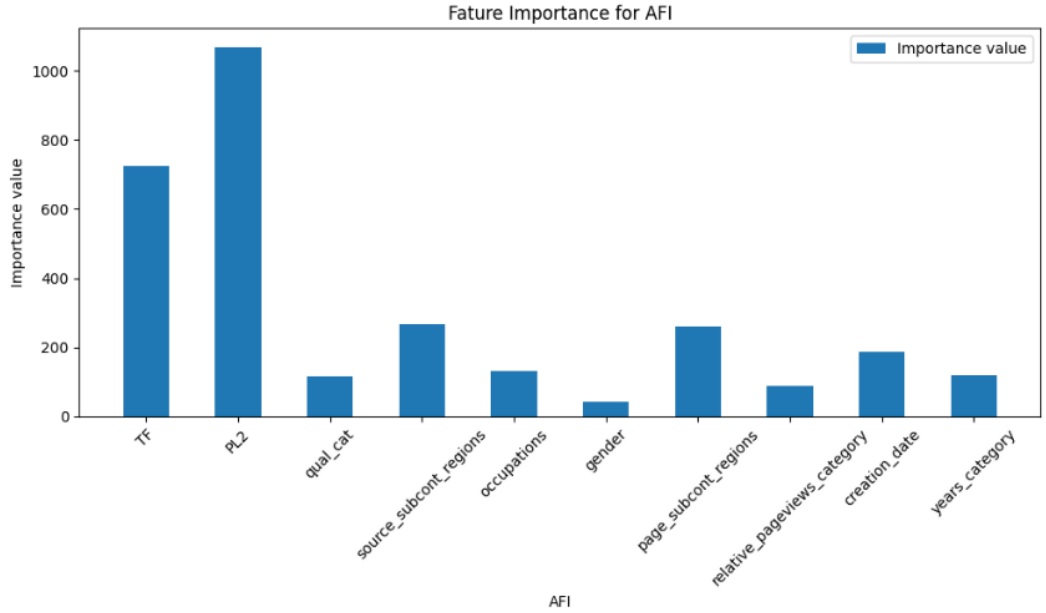


Figure 5.5: *This figure show the feature importances of the AFI features with PL2 being the most important feature and gender being the least important gender*

5.3 Evaluate Research Questions

This section will provide the answers to the questions posed in **Section 4.1**

5.3.1 Was fairness improved?

Firstly, we will toggle the question of :

What is the impact of incorporating LTR models on reducing bias and improving fairness in the ranking of the Wikipedia articles? In order to answer this question we will construct a general case Hypothesis in order to discuss if we have found critical evidence to suggest that our strategies have improved fairness in the ranking of Wikipedia articles.

The null hypothesis (H_0) and alternative hypothesis (H_1) are:

$$H_0 : FairnessinBaselineModels = FairnessProposedDesigns$$

$$H_1 : FairnessinBaselineModels < FairnessProposedDesigns$$

Using this hypotheses as the basis of all t-testings we can neglet all alterante hypothesis of

$$H_1$$

as there was no statistical significance to any of the strategies implemented in both relevance and fairness compared to the 2 baseline models.

5.3.2 Which type fairness feature performed best?

Secondly, we will answer the question of:

Which type of feature (e.g. Geolocation, Chronological, Article Author Attributes or all together) have the most significant impact on improving fairness for Wikipedia editors?

After evaluating feature importance and results, it has been determined that the geolocation features are the best performing features in terms of NRD, nDCG@10, and awrf. As shown in **Figure 5.2**, both geolocation features have similar importance in both strategies even after feature boosting, indicating their resilience and potential to influence the ltr in improving fairness for under-exposed groups.

On the other hand, features such as Author Article attributes have shown poor performance, potentially due to their combination of features. For instance, gender had little importance in both AAD and BIAAD strategies, and even after boosting the BIAAD, its importance decreased further as shown in **Figure 5.4b**. This suggests that other features, such as occupation, are more crucial and that gender may not be the most effective feature to improve fairness for under-exposed genders like females. Therefore also indicating that not all features perform the same in the ml process.

5.3.3 Feature Selection vs Feature Boosting

Thirdly, we will answer the question of:

Does feature selection or boosting of features have a the most positive impact?

To address this inquiry, we must refer to both **Figure 5.2** and **Figure 5.3**, which show that the boosting of certain features has resulted in a decrease in their importance, ultimately leading to a less equitable ranking. This phenomenon is also supported by the results presented in **Table 5.1**, where most of the feature selection strategies have exhibited better performance in both relevance and fairness, with the exception of the ACT and UACT strategies. Although the boosting strategy has outperformed the feature selection strategy in the latter two approaches, we still lack sufficient information to conclusively answer the question.

6 Conclusion

6.1 Summary

The goal of this dissertation is to address the issue of position bias in Wikipedia articles which results in certain protected groups being underrepresented in the top-n documents of a vertical ranked list. This problem has real-life implications such as countries that are overrepresented can benefit from economic growth. The Trec-Fair 2022 challenge served as the inspiration for this project, as it provided standardized data, metrics and evaluation techniques to construct the experiment. To improve fairness in ranking we proposed seven strategies that were implemented in PyTerrier using state-of-the-art Gradient Boosting machine learning algorithms (LightGBM). These strategies aimed to incorporate features that would enhance fairness without sacrificing relevance and were implemented using two approaches: feature selection and feature boosting. However, statistical significance was not observed within the relevance and fairness metrics, making it difficult to identify which approach was more effective. Nonetheless, the geolocation features showed the most promising results when used with feature selection, indicating that this approach may be more reliable for improving fairness at no cost to relevance. Although statistical significance was not achieved the experiment provided valuable insights into the Trec-Fair dataset which can be utilized to develop new strategies. By addressing the position bias problem in Wikipedia, this dissertation contributes to the broader effort towards promoting fairness and equity in information retrieval.

6.2 Reflection

We have successfully implemented 7 strategies aimed at improving both relevance and fairness, which suggests that there is a foundation for further research in these concepts. Our experiment was designed with careful consideration and we aimed to treat the outcomes as fairly as possible. However, we were unable to achieve statistical significance from our results which leaves us uncertain about the effectiveness of our strategies in improving fairness and relevance compared to baseline approaches. Upon reflection, we believe that tuning the hyperparameters of our machine learning algorithm is crucial in obtaining better results, even if this comes at the cost of computational resources. Additionally,having encountered a large number of unknown values in the dataset we should have developed an approach to handle this unknown data directly, which could have potentially improved the effectiveness of our strategies. Despite these challenges, we have gained valuable insights from our experiment which can guide future research in this field.

6.3 Future work

Our experiments have provided us with valuable insights into the Trec-Fair dataset, enabling us to develop strategies that can improve fairness and relevance. Our findings suggest that features based on geolocation are effective in enhancing both relevance and fairness, while features such as occupations also play a significant role in the machine learning algorithm. As a result, we propose implementing additional strategies that leverage these features to further enhance fairness in the

ranking system. Additionally, we believe that exploring the use of a variable number of relevance features, as proposed by Dai et al. (2011), could help to identify the optimal number of features required to achieve maximum performance in terms of relevance and fairness.

A Appendices

Bibliography

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J. and Wallach, H. (2018), 'A reductions approach to fair classification'.
URL: *https://arxiv.org/abs/1803.02453*

Al-Maskari, A., Sanderson, M., Clough, P. and Airio, E. (2008), The good and the bad system: Does the test collection predict users' effectiveness?, *in* 'Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval', Association for Computing Machinery, New York, NY, USA, p. 59–66.
URL: *https://doi.org/10.1145/1390334.1390347*

Biega, A. J., Diaz, F., Ekstrand, M. D. and Kohlmeier, S. (2019), Overview of the trec 2019 fair ranking track, *in* 'The Twenty-Eighth Text REtrieval Conference (TREC 2019) Proceedings'.

Caton, S. and Haas, C. (2020), 'Fairness in machine learning: A survey', *CoRR* **abs/2010.04053**.
URL: *https://arxiv.org/abs/2010.04053*

Collins, A., Tkaczyk, D., Aizawa, A. and Beel, J. (2018), 'A study of position bias in digital library recommender systems', **abs/1802.06565**.
URL: *http://arxiv.org/abs/1802.06565*

Craig Macdonald, N. T. (2020), 'Declarative experimentation in information retrieval using pyterrier', *Journal of Something* **10**.

Dai, J., Zhang, D., Fan, J. and Church, K. (2011), Compound word identification as chunking, *in* 'Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies', pp. 1427–1436.
URL: *https://www.microsoft.com/en-us/research/wp-content/uploads/2011/01/Dai2011.pdf*

Documentation, S. (2021), 'LGBMClassifier - lightgbm documentation', `https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMClassifier.html`. [Online; accessed 3-March-2023].

Dorleon, G., Megdiche, I., Bricon-Souf, N. and Teste, O. (2022), Feature selection under fairness constraints, *in* 'Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing', New York, NY, USA, p. 1125–1127.
URL: *https://doi.org/10.1145/3477314.3507168*

Ekstrand, M. D., McDonald, G., Raj, A. and Johnson, I. (2022), Overview of the trec 2021 fair ranking track, *in* 'The Thirtieth Text REtrieval Conference (TREC 2021) Proceedings'.

FAIR, T. (2022), 'Fair ranking 2022 participant instructions', `https://fair-trec.github.io/docs/Fair_Ranking_2022_Participant_Instructions.pdf`. Accessed: March 14, 2023.

Friedman, J. H. (2001), 'Greedy function approximation: A gradient boosting machine', *Ann. Statist.* **29**, 1189–1232.
URL: *https://doi.org/10.1214/aos/1013203451*

Grgić-Hlača, N., Zafar, M. B., Gummadi, K. P. and Weller, A. (2018), 'Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning', *Proceedings of the AAAI Conference on Artificial Intelligence* **32**(1).

Hardt, M., Price, E. and Srebro, N. (2016), 'Equality of opportunity in supervised learning'.

Joachims, T., Granka, L., Pan, B., Hembrooke, H., Radlinski, F. and Gay, G. (2007), 'Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search'.
URL: *https://doi.org/10.1145/1229179.1229181*

Kamishima, T., Akaho, S., Asoh, H. and Sakuma, J. (2012), Fairness-aware classifier with prejudice remover regularizer, pp. 35–50.

Li, G., Meng, X., He, Q., Chen, W. and Liu, T.-Y. (2017), Lightgbm: A highly efficient gradient boosting decision tree, *in* 'Advances in Neural Information Processing Systems', pp. 3149–3157.

Livemint (2018), 'A brief history of search', `https://www.livemint.com/Consumer/ZPKeAhhIljQa79t4OMQy2I/A-brief-history-of-search.html`. [Online; accessed 22-March-2023].

McFee, B. and Lanckriet, G. (2010), Metric learning to rank, *in* 'Proceedings of the 27th International Conference on International Conference on Machine Learning', ICML'10, Omnipress.

Nandy, P., Diciccio, C., Venugopalan, D., Logan, H., Basu, K. and Karoui, N. E. (2020), 'Achieving fairness via post-processing in web-scale recommender systems'.
URL: *https://arxiv.org/abs/2006.11350*

Oosterhuis, H. and de Rijke, M. (2020), 'Policy-aware unbiased learning to rank for top-k rankings', *CoRR* **abs/2005.09035**.
URL: *https://arxiv.org/abs/2005.09035*

Pedreshi, D., Ruggieri, S. and Turini, F. (2008), Discrimination-aware data mining, *in* 'Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', Association for Computing Machinery.
URL: *https://doi.org/10.1145/1401890.1401959*

Peyton Jones, S. (2017), How to write a great research paper, *in* '2017 Imperial College Computing Student Workshop, ICCSW 2017, September 26-27, 2017, London, UK', pp. 1:1–1:1.

Rai, V. (2020), 'Skewness and kurtosis in machine learning', *Medium* .
URL: *https://vivekrai1011.medium.com/skewness-and-kurtosis-in-machine-learning-c19f79e2d7a5*

Redi, M., Gerlach, M., Johnson, I., Morgan, J. T. and Zia, L. (2020), 'A taxonomy of knowledge gaps for wikimedia projects (first draft)', *CoRR* **abs/2008.12314**.

Sapiezynski, P., Budak, C. and Korolova, A. (2020), 'Fairness-aware attention mechanisms for graph convolutional networks', `https://github.com/sapiezynski/fairness-attention`. Accessed: 2023-03-30.

Sapiezynski, P., Zeng, W., Robertson, R. E., Mislove, A. and Wilson, C. (2019), 'Quantifying the impact of user attention on fair group representation in ranked lists', *CoRR* **abs/1901.10437**.
URL: *http://arxiv.org/abs/1901.10437*

Singh, A. and Joachims, T. (2018), 'Fairness of exposure in rankings', *CoRR* **abs/1802.07281**.
URL: *http://arxiv.org/abs/1802.07281*

Singh, D. and Singh, B. (2020), 'Investigating the impact of data normalization on classification performance', *Applied Soft Computing* **97**, 105524.
URL: *https://www.sciencedirect.com/science/article/pii/S1568494619302947*

Snoek, J., Larochelle, H. and Adams, R. P. (2012), 'Practical bayesian optimization of machine learning algorithms'.

Tonellotto, N., Macdonald, C. and Ounis, I. (2013), Efficient and effective retrieval using selective pruning, *in* 'Proceedings of the Sixth ACM International Conference on Web Search and Data Mining', Association for Computing Machinery, p. 63–72.
URL: *https://doi.org/10.1145/2433396.2433407*

Wikimedia Foundation (2021), 'Active editors by country - wikimedia statistics'. [Online; accessed 15-March-2023].
URL: *https://stats.wikimedia.org//en.wikipedia.org/contributing/active-editors-by-country/normal|map|last-month|(activity-level) 5..99-edits|monthly*

Wikimedia Foundation (n.d.), 'Where your money goes', `https://wikimediafoundation.org/support/where-your-money-goes/`. Accessed: March 29, 2023.

Woodworth, B., Gunasekar, S., Ohannessian, M. I. and Srebro, N. (2017), 'Learning non-discriminatory predictors'.
URL: *https://arxiv.org/abs/1702.06081*